8. CASE-BASED REASONING AND INFORMATION STRUCTURES IN COLLABORATIVE DESIGN: THE USE OF PRODUCT MODELS FOR INFORMATION EVOLUTION

Popova, Mina[1] Peter Johansson[2] Hans Lindgren[1]
Dept. of Visualization and Modeling. Dept. of Structural Engineering.
Chalmers University of Technology, 412 96 Gothenburg, Sweden

Abstract

The objective of our research project is to build up further knowledge about some aspects of the rational application of IT within collaborative design. We intend to integrate the platforms of case-based reasoning and information structures. Therefore, we especially concentrate on the integration of techniques in the form of professional applications: CAD-tools, word processors, and general applications and the World Wide Web, on one hand, and standards for the AEC industry: standardized product models such as IFC, national classification systems i.e. BSAB, on the other. A neutral format facilitating this integration has been found in XML dealing with both structured and weakly structured data produced and needed in design and construction projects. We focus on the designers' information needs and discuss a prototype of an everyday tool based on product- model and process-model technology. The most important role of this tool will be to promote efficient and flexible information management through case-based reasoning i.e. search, retrieval, storage, and reuse of information. If this process is made more effective, preferably automatic, it will motivate the design team to store valuable information concerning the whole lifecycle of an artefact. Through the use of standardized product models, this information will be sharable and suitable for the purpose of future reuse and feedback. The more often the design information is reused, the more general and adaptable it becomes and first then the designers will be able to state that they have achieved information evolution. This scenario requires, though, strengthening of the quality in the design office: a quality system for evaluating the information before it is reused, consequent use of standardized product models and IT. Only then the designers will become successful producers, re-users, and providers of information and will consolidate their position in the construction sector.

Keywords: *case-based reasoning, information structures, collaborative design and construction process.*

BACKGROUND

The gradual implementation of Information Technology (IT) during the last two decades has become the driving force behind the reengineering of the whole building industry. The need of information sharing during design, construction, and management has made the actors in the sector adopt new work methods. The fragmentary construction process is probably the main reason for low profits and quality problems with the production. The structure of the industry is also considered as one of the causes for the clear separation between the roles of architects and structural engineers taking place in the 20th century. Today, we have a rather distinct picture of two professional cultures with different languages (Björk et. al 1993 and Holgate 1986). Although we still consider architects and structural designers the main consultants in building, they have lost influence in the design and construction process. The architect is hired as a project consultant for a limited period, which decreases the insight, influence, and control over the project. Structural designers are in a similar position – they often enter a project even later than architects. This segmented process impedes these two professionals to receive appropriate feedback from the construction site and the facilities management perspective; thus, some of the knowledge gained when designing is lost and cannot be reused in future cases (Kalay 1997 and Lindgren 1992).

In the beginning, IT was associated with the automation of specific tasks and new techniques. Later, the phenomenon 'islands of automation' (Fruchter and Clayton 1993) and topics such as: integration, communication, case-based design systems, shared product and information models were brought to light. Today, when we look at the buildings we have designed, we should ask ourselves what was done well and what we ought to change. The answers to these questions are derived analysing the client's needs and matching them with the experience of the designers. To an increasing extent, these answers are conceptualised during a pre-project stage of the design and construction process. By integrating these issues in the design process, the professionals will be able to add considerable value to the client. From the designers' point of view, this could be achieved through information systems supporting the re-use of design information.

THE USE OF IT IN CONSTRUCTION TODAY

Since the 1980-ies, most industries have investigated how IT can add value to the final product, and the construction sector is no exception. It is impossible, though, to adopt directly concepts from other sectors such as the manufacturing industry because building seldom witnesses serial production of an artefact while this is the essence of manufacturing. The construction business is already aware of the benefits of sharable information structures, case-bases for project information, and international standards. In order to profit from the evolving modelling standards in, more information about a particular artefact has to be generated earlier in the process compared to traditional building design.

In the 1990-ies, the focus was mainly on product models, this partly leading to international efforts with standards like ISO 12006-2 (by the International Standardization Organisation) and the industry initiative Industry Foundation Classes (IFC) triggered by the International Alliance for Interoperability. Usually, architects are the prime users of models to visualise the design concept when presented to clients and users. In some cases, HVAC and structural engineers also build models in order to simulate different functions of the building. Despite all discussions, we still see very few examples from the construction industry using this type of methodology. Apart from the problems with defining the standards for information exchange, it is a very expensive method due to the limited use of the models considering the whole project. We believe that the situation can improve if more value is added to the integrated models and case-bases by widening their use and reuse, thus promoting information evolution in more than one project.

Considering all said above, we could state that the building industry can profit from using the evolving modelling standards and case-based design (CBD)-systems. The present situation increases the demands on the architects and the structural designers, especially in their role as information suppliers in the building process. These new options should be discussed in the context of an integrated, collaborative design process.

THE RESEARCH PROBLEM

Our research focuses on building up further competence about the rational application of IT in collaborative design and, especially, on methods to join the design efforts of architects and structural engineers as early as possible in an integrated process. At present, we investigate the integration of the platforms of information structuring and CBD (Johansson 2000). The core of our concept is to use the evolving product models in order to structure design information and together with the CBD methodology to search, retrieve, reuse, and evolve the information. In this way, we can bridge the gap between the modelling standards and the future use of IT in design. Probably, the most significant feature of CBD is that it motivates the designers to save and structure useful and reusable information. The next step would be to point out possible ways towards a higher degree of integration between CBD-systems and information models and structures. The concepts in the present project are based on our previous participation in the cross-disciplinary effort *Project Wide Databases* carried out at Chalmers, where the work has resulted in two licentiate theses in the fields of CBD (Johansson 1996) and information structures (Popova 1997), respectively.

CASE-BASED REASONING/CASE-BASED DESIGN

Case-based reasoning (CBR) originates from the cognitive observation that humans often rely on past experience to solve new problems. Schank (1982) created a model describing how case-specific information can be stored in a memory and retrieved when needed. The same knowledge structure is used in remembering, understanding, experiencing, and learning, and it changes as a result of its experience. This model has gradually evolved focusing on indexing and storing cases and together with the issue of adapting a case to a new situation this became the area of CBR (Kolodner 1993). Because of the importance of experience in design, many CBD-systems have been implemented for problem solving (Maher et al. 1997). The handling of small but difficult problems has proved that case-specific information is usable and CBR ought to be a part of a design system (Hartvig 1999). This becomes clearer as the required information about the artifacts in building projects increases constantly and the need to reuse information from old cases grows stronger.

BRIDGEBASE – A Simple CBD-System

In BridgeBase (Edlund et al. 1993), CBR is used in the preliminary design of highway bridges by comparing the bridge designed (new bridge) with built bridges (old bridges). The ART-IM development system (ART-IM 1992) was used to create the prototype.

8.1.1 Representation

The information about 215 highway bridges of various types was obtained from a publication of the Swedish National Road Administration (Vägverket). This information was stored in an object-oriented manner where for every old bridge in (Vägverket 1988) an instance of the class *common-bridge* was created (Fig. 1).

```
(defschema B-P1075
            (instance-of common-bridge)
            (bro-id "P 1075")
            (bridge-type "Concrete frame bridge")
            (name "Crossing road 634 Kinnarumna")
            (total-length 21.3)
            (width 13.0)
            (support-angle 70)
            (radie)
            (c-f-h 0.72)
            (c-s-h 1.3)
            (vot "1/5")
            (number-of-beams)
            (width-of-beams)
            (diameter)
            (reinforcement "Non-prestressed")
            (number-of-spans 1)
            (number-key 1))
```

Figure 1. *A Case in BridgeBase.*

Retrieval

BridgeBase retrieves information about old bridges relevant to the new design. First, the similarity for every member variable of the class is calculated separately, depending on the variable type. For a numerical variable, a match function is used (Fig. 2).

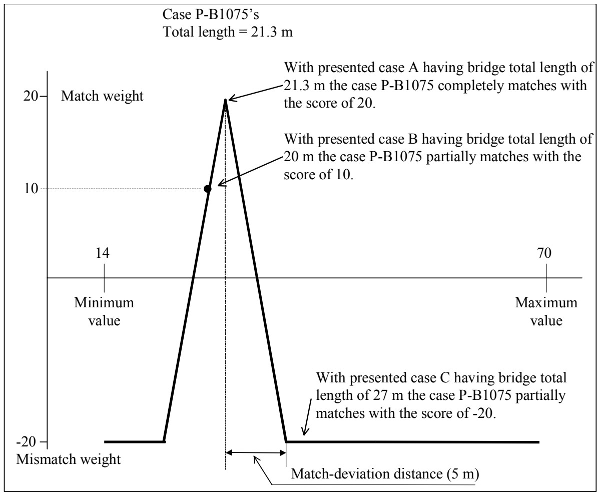


Figure 2. *Match function for numerical variables in ART-IM.*

It is possible to set weights on every member variable of the class since they have different importance. The similarity between an old bridge and a new one is calculated as the weighted sum of the match score for the variables:

$$CaseMatchScore = \frac{\sum VariableMatchScore \times VariableWeight}{\sum VariableWeight}$$

Equation 1. *Calculation of the match score.*

A Design Example

In this very simplified example, BridgeBase is used to select a suitable type for a 45m long bridge. As the information Total–length = 45m is entered into the presented case, BridgeBase matches and prints out information about the five most similar bridges:

Match score	Bridge type	Total length (m)	Number of spans
0.0260	Concrete slab bridge	45.2	3
0.0251	Concrete frame bridge	44.4	2
0.0249	Concrete slab bridge	44.2	3
0.0249	Concrete slab bridge	45.8	3
0.0238	Concrete frame bridge	44.0	2

Table 1. *Match results.*

BridgeBase "recommends" the bridge types *concrete slab bridge* and *concrete frame bridge* in two or three spans (Table 1). This allows the designer to draw some conclusions or key in more information and conduct another CBR-session. We can conclude that CBR supports the reuse of experience in the field of structural engineering and that the more old cases the case-base contains the better the functionality of the CBD-system.

INFORMATION STRUCTURES CONCERNING CBD

If we consider the architectural artefact as a part of an extremely complex reality, the need of a model, a representation of the artefact is obvious. Today, the construction community has access to standard product models like ISO 12006-2 and IFC, and national classification systems such as the BSAB-system in Sweden (BSAB-96 1999). The ISO standard creates a common frame in construction classification to support information exchange e.g. in CAD, specifications, and cost estimate (ISO 1997). It is based on a process model identifying resources, processes, and results, and supporting design, production, and use. Since ISO 12006-2 does not contain classification tables, it recommends the use of regional and national ones like the BSAB-system.

The IFC model is a framework of classes facilitating the information exchange between computer systems, the goal being software interoperability in the AEC/FM domain. This implies that all software carrying information about an artefact and about the processes in its lifecycle ought to handle IFC. The further development of IFC requires the cooperation between different interest groups and takes place in stages; the results should be carefully scrutinised before they are included in the model. Although national/regional classification systems do vary, they are significant for the information exchange in cost estimates, specifications, databases, CAD, libraries. Due to their long traditions (since 1934 in Sweden), these systems are well established today and have evolved: for instance, the third generation BSAB-system – BSAB 96 complies fully with the ISO 12006-2 standard.

In our opinion, standard product models like ISO 12006-2 and IFC cannot be directly used in their present form by design teams. These standards serve as a foundation for the product model in a CBD-system but they contain product information as the top-level concept (Turk 1998) and are too comprehensive (Popova 1997). In some respects, they lack the necessary flexibility to support all the stages in the design process (Ekholm and Tarandi 2000).

Design rationale (intent), is the rationale behind the decision-making process in design and the collection of information about the evolution of a design product (de la Garza and Oralkan 1995). It is also defined as a type of knowledge that facilitates CBD. A conceptual framework for the AEC industry should be developed in order to represent this deep knowledge (Simoff and Maher 1998). The fact that concepts like *Building section* and *Frame*, vital for the design rationale in CBD, are not yet defined in IFC, renders the standard product models incapable of acting as a conceptual framework at present. We believe that the overall goal of the product models should be to assist the activities of the design team (Popova 1997, and Turk1998). It should also allow the designers to create new types of objects (Fisher 1994) and to evolve their information structure.

Structured, Weakly Structured, and Raw Data

A common problem when using integrated information structures and CBD-systems is how to acquire the information needed. The information used by designers is divided into three categories (Simoff and Maher 1998):

1. *Structured data,* e.g. information covered by the standardized product models and created by applications promoting their use;
2. *Weakly structured data,* e.g. information not covered by the standardized product models or information created by applications irrelevant to the product models;
3. *Raw data,* e.g. raster and animated images, sketches, audio and video data.

Other examples of structured data are: attribute-value pairs, relational tables, object-oriented structures, and classification tables. Much of the information produce by designers is weakly structured data: texts in free or table format, calculation documents. Design calculation documents contain information unavailable elsewhere: concepts (e.g. frame) and information about the solution chosen for a specific function of a building. Here, we find information about the design process leading to a solution and, to some extent, the design rationale in the process. Architects often store information as raw data. Standard information models like IFC deal mostly with the first category while CBD-systems need information from all the groups.

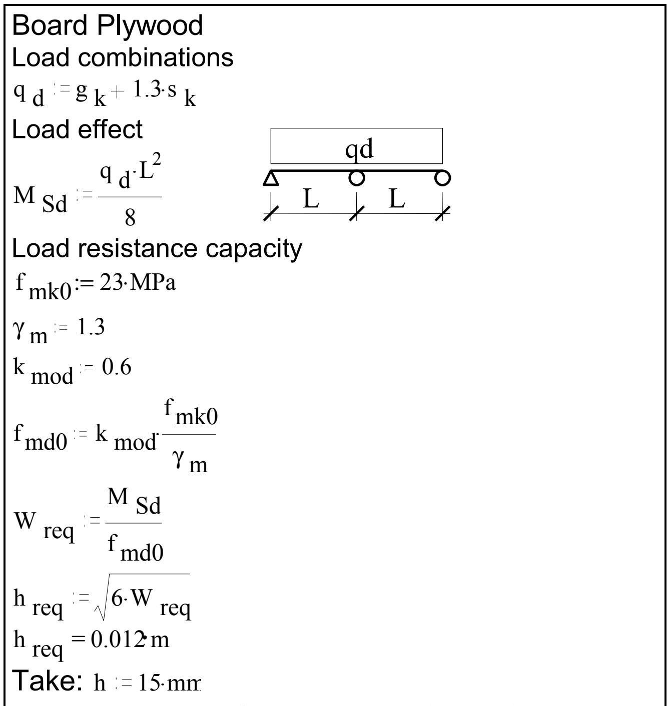
As stated earlier, the main purpose of models like IFC is transferring information between applications. This also enables the acquiring of information into a case-base where CBR-sessions can be conducted. The information retrieved from the CBD-system is in the format of the product models and can be imported into other applications, which facilitates reuse. Consequently, product models should be the first choice in representing structured data. For weakly structured data, the task is more complicated.

ARCADE

ARCADE, a prototype developed and implemented at Chalmers, investigates methods how to capture, retrieve, reuse, and evolve weakly structured data in design calculation documents and how to automate this process. Another goal has been to investigate how the information about the design process in these documents could be used in CBD. ARCADE acquires its cases from and produces results in the format of Mathcad 6.0.

Representation and Acquisition in ARCADE

A design calculation document can be regarded as a list of variable definitions with a name, a physical unit, and a value. Besides variable definitions, it can contain pictures and comments. The variables can be independent or dependent. When the variable names are used in a definition of a dependent variable, a dependency is created between them, e.g. between q_d and g_k, leading to a dependency structure. The document is subdivided into sections by headings on different hierarchical levels. ARCADE acquires cases by using the variable definitions, the dependency structure, and the headings.


Board Plywood

Load combinations

$q_d := g_k + 1.3 \cdot s_k$

Load effect

$M_{Sd} := \frac{q_d \cdot L^2}{8}$

Load resistance capacity

$f_{mk0} := 23 \cdot MPa$

$\gamma_m := 1.3$

$k_{mod} := 0.6$

$f_{md0} := k_{mod} \cdot \frac{f_{mk0}}{\gamma_m}$

$W_{req} := \frac{M_{Sd}}{f_{md0}}$

$h_{req} := \sqrt{6 \cdot W_{req}}$

$h_{req} = 0.012 \cdot m$

Take: $h := 15 \cdot mm$

Figure 3. *A part of a design calculation document.*

A Design Example

In this example, the case-base contains design calculation documents in Swedish from six projects done by different designers in Gothenburg (cf. Fig. 3 and Fig. 4).

The example concerns the foundation of a warehouse where most of the conceptual design has been carried out, the columns have been placed, and a piled foundation has been chosen.

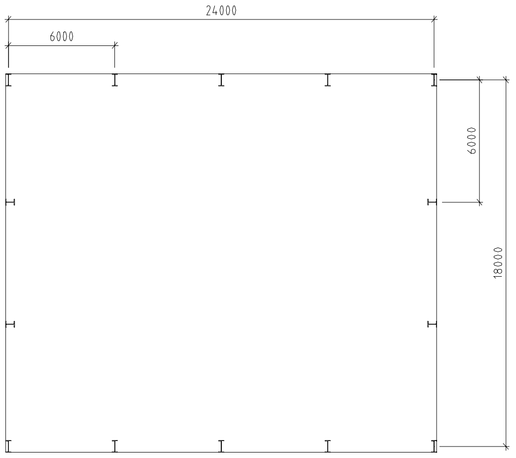
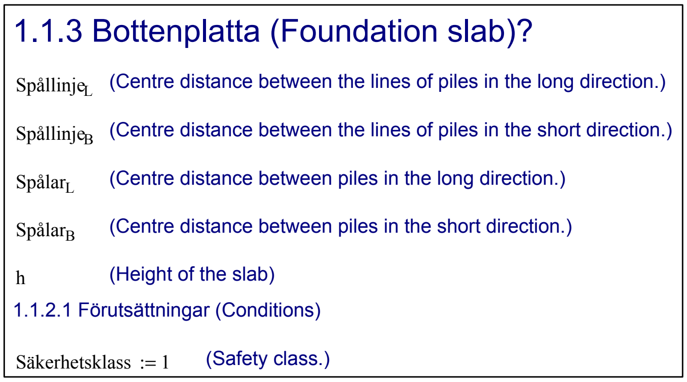


Figure 4. *Column layout.*

These design calculation documents contain information about the main geometry and the functions of the building as well as the load calculations. As the structural engineer starts designing the foundation, he/she describes the known and the wanted variables (Fig.5).



1.1.3 Bottenplatta (Foundation slab)?

Spållinje_L (Centre distance between the lines of piles in the long direction.)

Spållinje_B (Centre distance between the lines of piles in the short direction.)

Spålar_L (Centre distance between piles in the long direction.)

Spålar_B (Centre distance between piles in the short direction.)

h (Height of the slab)

1.1.2.1 Förutsättningar (Conditions)

Säkerhetsklass := 1 (Safety class.)

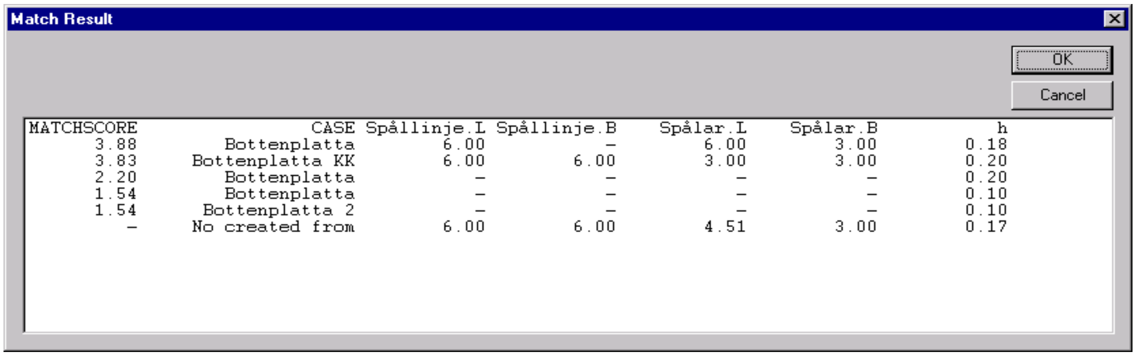
Figure 5. *The section of the calculation document design problem.*

Now a retrieval session is performed with match results as shown below in Fig. 6. ARCADE matches only relevant old cases where the heading describes the class (concept) of a section. In Fig 6., five different old cases of the class *Foundation Slab (Bottenplatta)* have been retrieved. In order to describe this process let us focus on the two most similar cases: *Foundation Slab* and *Foundation Slab KK (BottenplattaKK).* In ARCADE*,* more than just the similarity of the member variables of the components is matched (as in BridgeBase).
After the matching, two main questions must be answered:

1. *Does the old case contain the information we need?*
2. *Does the current document contain the variables that the old case needs defined?*

The first question is answered by a goal state match; the second – by a footprint match.
In the former, the design problem searches the old cases for the necessary information. As seen in Fig. 6, the only difference between these old cases is that *Foundation Slab KK* contains $Spållinje_B$ while *Foundation Slab* does not, giving *Foundation Slab KK* a higher goal state match score. In the footprint match, each old case looks in the current design calculation for the information needed: i.e. the variables used but not defined in the old case (footprint variables). ARCADE determines the footprint variables of an old case by the dependency structure at acquisition time. The two cases have the following footprint variables:

Foundation Slab** (Bottenplatta):** q_{NLk} (Live load) and S_L (Centre distance of the columns in the long direction).
Foundation Slab KK** (Bottenplatta KK):** q_{NLk}, S_L and S_B (Centre distance of the columns in the short direction).



```
MATCHSCORE               CASE Spållinje.L Spållinje.B    Spålar.L    Spålar.B           h
      3.88       Bottenplatta        6.00           -        6.00        3.00        0.18
      3.83    Bottenplatta KK        6.00        6.00        3.00        3.00        0.20
      2.20       Bottenplatta           -           -           -           -        0.20
      1.54       Bottenplatta           -           -           -           -        0.10
      1.54     Bottenplatta 2           -           -           -           -        0.10
         -    No created from        6.00        6.00        4.51        3.00        0.17
```

Figure 6. *The match results.*

If the footprint variables from a new case are found in the present design document, the similarity is calculated as that of a member variable. We achieve a better retrieval of available information about the design process by using the footprint variables in the match process.

The value of q_{NLk} in the case *Foundation Slab* gets a higher match score than in *Foundation Slab KK* since it corresponds better to the variable value in the current case. If the structural designer now chooses to reuse the values retrieved and/or the design process (calculations) of the old case, he/she also has the opportunity to evolve them and make them more reusable in the future. In addition, the system registers how many times an old case has been reused, which is seen as a measure of a case's generality.

Evaluating ARCADE, we can conclude that weakly structured information can be acquired automatically during ordinary design work in order to be used for CBR-sessions. Also, accessing information about the process in CBD supports the reuse and evolution of the design process itself as well as the retrieval and the adaptation of design information.

XML AS A COMMON FORMAT FOR CBD

ARCADE shows that it is possible to achieve a more or less automatic CBD-process incorporated into the application producing the design information. But usually, a CBD-system needs information from different applications, which complicates the acquisition process and requires an independent format. Most applications export data in the ASCII-format that can be used by a CBD-system: e.g. this is how ARCADE acquires information from Mathcad. Unfortunately, this format often lacks something: in Mathcad, the values of the calculated variables are not stored in the ASCII-format. The SAM-system (Maher 1997) attempts to overcome such problems by the rational use of the World Wide Web (WWW) and the HyperText Markup Language (HTML) format. Some of the advantages are that most applications can handle the HTML-format, that the linking of documents is easy, and that there are search functions. But, the format can represent only a limited number of information types – equations created in Mathcad are stored as pictures.

We believe that the evolving Extensible Markup Language (XML) can serve as an information exchange format between most applications since it handles both structured and weakly structured data. The XML-format is recommended by the W3C as "the universal format for structured documents and data on the Web" <www.w3c.org/XML/>, <www.w3c.org/Consortium/>. It is described as "a set of rules, […] for designing text formats for data" to produce unambiguous files, easy to generate and read (by a computer), and characterized by extensibility, support for internationalization/localization, and platform-independency <www.w3c.org/XML/1999/XML-in-10-points/>. The XML-format is used in our work because it is neutral and flexible and allows structured data in Web-applications by letting the user define own tags to structure (tag) data as it is collected from various places and integrated using a browser. The Mathematical Markup Language (MathML) is an XML-application for describing and capturing the structure and content of mathematical notations <www.w3c.org/TR/MathML2>. Today, the development goes on and aims at including a query language in the XML-standard. Since XML can represent both structured and weakly structured data, it is logical to conclude that it will gradually replace the ASCII- and the HTML-format and become a universal standard.

The XML-format is suitable for the representation of the IFC product models too: the work with mapping EXPRESS-schemas and corresponding data to XML progresses (Liebich 2000). The prototype developed in the project 'Application of IFC in Sweden ' proves that classification codes can be attached to a CAD-model and the information can be translated into the IFC Part21- and XML-format (Ekholm and Tarandi 2000). The prototype BSAB Demonstrator has shown that national classification tables can also be structured with XML-tags and project documents written in MS Word (Microsoft Corporation) can be exported as XML-documents (Häggström et al. 2000). The information in the classification systems complements the product models and when translated into a neutral format they can be used together in CBR-sessions. Fig. 7 illustrates our vision of the representation of design information using XML as a common platform:

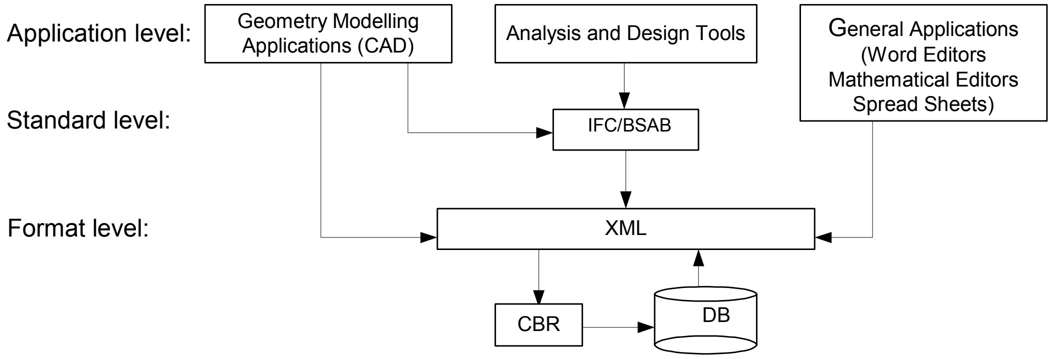
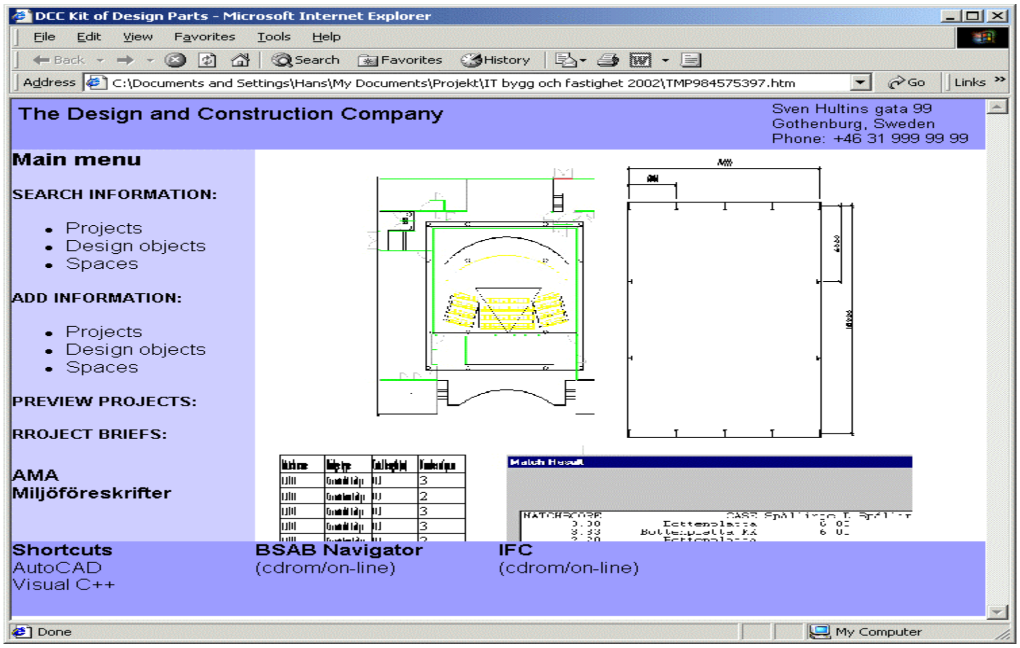


Figure 7. *Representation of design information*

Structure of the Kit of Design Parts

In our work, we are not contributing by creating new techniques but rather by developing applications based on available technology and standards. Our profile can be described as the interaction and the interface between existing standards: IFC and BSAB; Web-technique: Apache, XML, images, and a Microsoft IE5.5 browser; programming languages: C++, PHP4, and JavaScript; and professional applications: CAD-files, text files, calculation documents. A part of the software we have used (Apache, PHP4, JavaScript) is open source, characterised by neutrality, platform-independency, and flexibility. According to our experience, even persons with limited computer skills are able to learn the basics and understand the concepts behind such prototypes. The outlined background points to a new way of understanding and dealing with structured, weakly structured and raw data by integrating different methods. It gives us powerful tools to build standardised and yet user-friendly design information systems.



The Design and Construction Company

Sven Hultins gata 99
Gothenburg, Sweden
Phone: +46 31 999 99 99

Main menu

SEARCH INFORMATION:

- Projects
- Design objects
- Spaces

ADD INFORMATION:

- Projects
- Design objects
- Spaces

PREVIEW PROJECTS:

RROJECT BRIEFS:

AMA
Miljöföreskrifter

Shortcuts
AutoCAD
Visual C++

BSAB Navigator
(cdrom/on-line)

IFC
(cdrom/on-line)

Figure 8. *The interface of the Kit of design parts.*

The prototype Kit of Design Parts is at present organised as a file structure (directory) placed on an Apache server. The user starts from an index file and navigates through the archive in search of projects, design objects, spaces, and project previews. This function is programmed in PHP4 and JavaScript. The design information is stored as XML-files and images, which allows editing, searching, and addition of new data through HTML-forms.

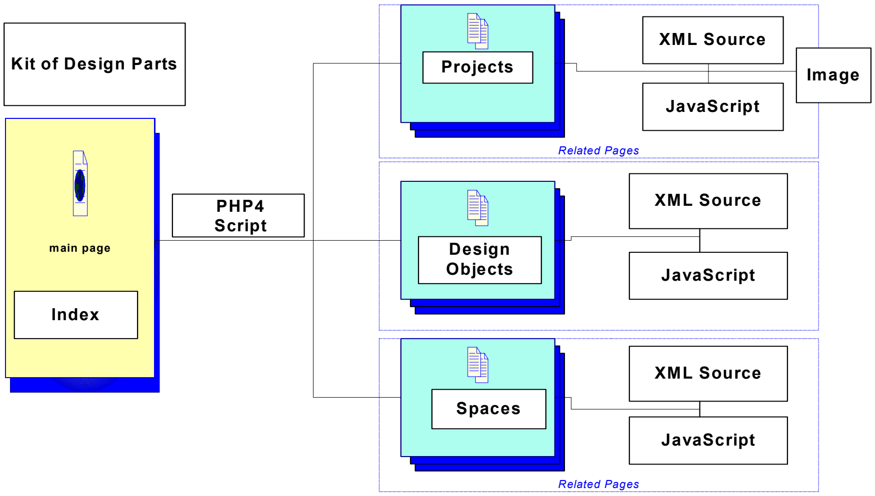


Figure 9. *Basic structure of the Kit of Design Parts.*

The flexibility of the system is expressed through the various categories of information: references to national classification systems (BSAB 96), international standards (IFC), and more personal information defined by the design team: project title, key words, designer's notes, ID number, etc. Here, we can store and search for both general project information: text documents, project plans, multimedia data and specific project information i.e. design objects:

windows, doors, or spaces, often of particular interest to architects during the design process. Our next attempt is to develop the prototype in such a way that changes/improvements in the design will be easily captured thus updating the XML source documents. The user will then be able to retrieve evolved information that he/she can reuse in the ongoing project or in future projects. In short, we will be able to perform CBR-sessions by using weakly structured data produced by architects. The final step in our project will be to combine the two prototypes – ARCADE and Kit of design parts into one CBD-system accessible through a common interface. In this way, we will achieve information evolution in the design and construction process through CBR methodology (Fig.10).

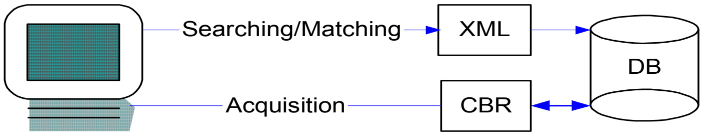


Figure 10. *The concept of informatin evolution through the design and construction process.*

CONCLUSIONS

We can conclude that the concepts of product models and CBR have the potential to bring about a more efficient structuring of information in the construction process. Designers should extend their use of integrated product models in order to become not only the producers but also the re-users of the information. If the systematic use of models and CBD-systems becomes a common practice, it will promote the information evolution in the design process, the storage and retrieval of information becoming a natural part. This change will, in turn, affect the culture of the entire construction business. In short, the implementing of new work methods and routines is indispensable if the transition from a sequential building process to an integrated, collaborative one is to succeed.

REFERENCES

ART-IM 2.5 (1992) Reference Manuals. Inference Corporation, Los Angeles.

Björk B. C., Huovila P. & Hult, S. (1993) Integrated Construction Project Document Management (ICPDM). In EuropIA'93 Conference.

BSAB 96 System och tillämpningar, (SR-Rekommendationer 10)(1999). AB Svensk Byggtjänst, Stockholm.

de la Garza J. M. and Oralkan G. A. (1995). Using design Intent for Interpreting Brand-Name-or Equal Specification. Journal of Computing in Civil Engineering, Vol. 9, No. 1, pp. 43-56.

Edlund B., Löfquist P., Johansson P. (1993) Case-based reasoning in bridge design. EG-SEA-AI Workshop "Application of Artificial Intelligence in Structural Engineering", Lausanne, Switzerland, pp. 60-61.

Ekholm A. and Tarandi V. (2000). Application of IFC in Sweden – phase 2. Final report. The Swedish Building Centre Systematics, Stockholm.

Fischer G. (1994). Domain-Oriented Design Environments. In "Automated Software Engineering", Kluwer Academic Publishers, Boston, MA, pp. 177-203.

Fruchter R., Clayton M., et al. (1993) Interdisciplinary Communication Medium for Collaborative Design. The Third International Conference AI CIVIL-COMP93. Edinburgh.

Gero J. S. (1993) AI and Computational Models of Design, IABSE Colloquium " Knowledge-Based Systems in Civil Engineering" Beijing, China, pp. 25-32.

Holgate A. (1986) The art in structural design. Oxford University Press.

Häggström L., Johansson B., Rosengren P. (2000). BSAB Demonstrator – etapp 2. Final report. The Swedish Building Centre Systematics, Stockholm.

Hartvig Susanne C. (1999) A Framework for IT-based Design Tools – Enabling Integration and Design Support. Report 6 Department of planning, Technical University of Denmark.

ISO (1997). ISO/DIS 12006-2 Organisation of information about construction works – Part 2: framework for classification of information. Geneva: International Standardization Organisation.

Johansson P. (1996) Case-Based Structural Design – reusing design calculation document information. Chalmers University of Technology, Division of Steel and Timber Structures, Publ. S 96:3, Göteborg.

Johansson P. (2000) Case-Based Structural Design – using weakly structured product and process information. Chalmers University of Technology, Division of Steel and Timber Structures, Publ. S 00:7, Göteborg.

Kalay Y. E. (1997) P3: An Integrated Environment to Support Design Collaboration. In Representation and Design, ACADIA, Cincinnati, Ohio.

Kolodner J. (1993) Case-Based Reasoning. Morgan Kaufmann, San Mateo.

Liebich T. (2000) XML schema language binding of EXPRESS for ifcXML, MSG-01-001(Rev 3), International Alliance for Interoperability.

Lindgren H. (1992) A Framework for Computer-aided Reuse of Architectural Design Solutions. CIB W78 workshop on Computer-Integrated Construction, 12-14th of May, Montreal, Canada.

Maher Mary Lou, Pu Pearl (1997) Introduction to the Issues and Applications of Case-Based Reasoning in Design. In Issues and Applications of Case-Based Reasoning in Design, Maher and Pu (editors), Lawrence Erlbaum Associates, Mahwah, NJ, ISBN 0-8058-2312-3. pp. 1-10.

MathSoft (1995) Mathcad User's Guide Mathcad 6.0. Cambridge, MA, MathSoft, Inc.

Popova M. (1997) IT and the Architect's Role. Division of Design Computing, Chalmers University of Technology, CTH-A CADLAB 1997:1, Gothenburg.

Schank R. (1982) Dymanic Memory: A theory of learning in computers and people, Cambridge University Press.

Simoff S. and Maher M.L. (1998) Ontology-based multimedia data mining for design information retrieval, *Proceedings of ACSE Computing Congress*, Cambridge.

Turk Ziga (1998) On Theoretical Background of CAD. Lecture Notes in Artificial Intelligence, 1454, Springer, ISBN 3-540-64806-2, pp. 490-496.

Vägverket (1988) Brotyper Tabeller. (Bridge types. Tables, In Swedish). Vägverket, Borlänge.